IN THE CLAIMS

In response to the Examiner's Amendment in the Notice of Allowance mailed 11/14/2008, please amend the claims as follows:

1. (Previously Presented) An operating system identification system including a node capable of executing computer code comprising:

an identification module configured to execute a plurality of operating system identification tests, each operating system identification test configured to make an identification of an operating system being executed by a network node;

a plurality of identification rules configured to define a procedure by which the identification module makes an overall identification of the operating system, wherein the overall identification is based at least in part on at least one of the identifications made by the plurality of operating system identification tests; and

a conflict resolution module configured to detect at least one of a plurality of cases defined by a plurality of conflict resolution definitions in which at least some of the plurality of operating system identification tests disagree in their identification of the operating system, and configured to, upon detecting such a case, to make an identification of the operating system and to cause the identification module to modify the overall identification based at least on the identification made by the conflict resolution module;

wherein a confidence level is assigned to the identification of the operating system based on a predetermined confidence level stored in association with at least one of a plurality of identification fingerprints used to identify the operating system;

wherein the identification of the operating system by one of the operating system identification tests is dependent on the identification of the operating system by another one of the operating system identification tests;

wherein a list of open ports on the network node is generated and, based on the list of open ports, another identification of which operating system is executed by the network node and another confidence level indicating a degree to which the other identification is deemed accurate are generated, wherein making the overall identification of the operating system is further based on the other identification and the other confidence level;

wherein generating the list of open ports comprises retrieving a previously constructed list of open ports.

2. (Original) The operating system identification system of Claim 1, wherein the plurality of operating system identification tests includes a Transmission Control Protocol identification test.

3. (Original) The operating system identification system of Claim 2, wherein the plurality of operating system identification tests further includes an Internet Control Message Protocol identification test.

4. (Original) The operating system identification system of Claim 3, wherein the plurality of operating system identification tests further includes a banner matching test.

5. (Original) The operating system identification system of Claim 4, wherein the plurality of operating system identification tests further includes an open port signature test.

6. (Original) The operating system identification system of Claim 5, wherein the plurality of operating system identification tests further includes a NULL session enumeration test.

7-10. (Cancelled)

11. (Previously Presented The operating system identification system of Claim 4, wherein each identification fingerprint is configured to associate an operating system with responses expected to be generated by the associated operating system in response to execution of one of the identification tests, wherein the identification made by each

identification test is based, at least in part, on comparisons between the identification fingerprints and actual responses generated by a tested operating system in response to execution of one of the identification tests.

12. (Original) The operating system identification system of Claim 11, further comprising a logic engine, wherein the logic engine performs the comparisons between the identification fingerprints and actual responses.

13. (Original) The operating system identification system of Claim.12, wherein at least one of the comparisons performed by the logic engine is a fuzzy logic comparison.

14. (Previously Presented) The operating system identification system of Claim 4, wherein each identification of the operating system made by one of the identification tests is associated with the confidence level indicating a degree to which the identification is deemed to be accurate, and wherein the overall identification is further based on the confidence level associated with the at least one identification relied upon to make the overall identification.

15. (Original) The operating system identification system of Claim 14, wherein each associated confidence level represents a probability that the identification is accurate.

16. (Previously Presented) An operating system identification system including a node capable of executing computer code comprising:

an identification module configured to execute a plurality of operating system identification tests including at least a Transmission Control Protocol identification test, an Internet Control Message Protocol identification test, and a banner matching test, each operating system identification test configured to make an identification of an operating system being executed by a network node; and

a plurality of identification rules configured to define a procedure by which the identification module makes an overall identification of the operating system, wherein the overall identification is based at least on at least one of the identifications made by the plurality of operating system identification tests;

wherein a confidence level is assigned to the identification of the operating system based on a predetermined confidence level stored in association with at least one of a plurality of identification fingerprints used to identify the operating system;

wherein the identification of the operating system by one of the operating system identification tests is dependent on the identification of the operating system by another one of the operating system identification tests;

wherein a list of open ports on the network node is generated and, based on the list of open ports, another identification of which operating system is executed by the network node and another confidence level indicating a degree to which the other identification is deemed accurate are generated, wherein making the overall identification of the operating system is further based on the other identification and the other confidence level;

wherein generating the list of open ports comprises retrieving a previously constructed list of open ports.

17. (Original) The operating system identification system of Claim 16, wherein the plurality of operating system identification tests further includes an open port signature test.

18. (Original) The operating system identification system of Claim 17, wherein the plurality of operating system identification tests further includes a NULL session enumeration test.

19. (Previously Presented) The operating system identification system of Claim 16, wherein each identification fingerprint is configured to associate an operating system with responses expected to be generated by the associated operating system in response to execution of one of the identification tests, wherein the identification made by each identification test is based, at least in part, on comparisons between the identification fingerprints and actual responses generated by a tested operating system in response to execution of one of the identification tests.

20. (Original) The operating system identification system of Claim 19, further comprising a logic engine, wherein the logic engine performs the comparisons between the identification fingerprints and the actual responses.

21. (Original) The operating system identification system of Claim 20, wherein at least one of the comparisons performed by the logic engine is a fuzzy logic comparison.

22. (Previously Presented) The operating system identification system of Claim 16, wherein each identification of the operating system made by one of the identification tests is associated with the confidence level indicating a degree to which the identification is deemed accurate, and wherein the overall identification is further based on the confidence level associated with the at least one identification relied upon to make the overall identification.

23. (Original) The operating system identification system of Claim 22, wherein each associated confidence level represents a probability that the identification is accurate.

24. (Currently Amended) A method of identifying an operating system executed by a network node, comprising:

transmitting a first plurality of Transmission Control Protocol packets to a network node on a computer network, receiving in response a second plurality of Transmission Control Protocol packets, and generating, based on characteristics of the second plurality of Transmission Control Protocol packets, a first identification of which operating system is executed by the network node and a first confidence level indicating a degree to which the first identification is deemed accurate;

transmitting at least a first plurality of Internet Control Message Protocol packets to the network node, receiving in response at least a second plurality of Internet Control Message Protocol packets, and generating, based at least on characteristics of the second plurality of Internet Control Message Protocol packets, a second identification of which operating system is executed by the network node and a second confidence level indicating a degree to which the second identification is deemed accurate;

connecting to at least one open port on the network node, transmitting to the at least one open port data configured to cause the at least one open port to return at least one banner, and generating, based on the at least one banner, a third identification of which operating system is executed by the network node and a third confidence level indicating a degree to which the third identification is deemed accurate; and

generating an overall identification, based on at least the first identification, the first confidence level, the second identification, the second confidence level, the third identification, and the third confidence level, of the operating system executed by the network node;

wherein the first confidence level is assigned to the first identification of the operating system, the second confidence level is assigned to the second identification of the operating system, and the third confidence level is assigned to the third identification of the operating system based on a predetermined confidence level stored in association with at least one of a plurality of identification fingerprints used to identify the operating system;

wherein the first identification of the operating system, the second identification of the operating system, and the third identification of the operating system by one of a plurality of operating system identification tests are dependent on the identification of the operating system by another one of the operating system identification tests;

wherein a list of open ports on the network node is generated and, based on the list of open ports, a~~nother~~ fourth identification of which operating system is executed by the network node and a~~nother~~ fourth confidence level indicating a degree to which the fourth identification is deemed accurate are generated, wherein making the overall identification of the operating system is further based on the [[other]]fourth identification and the [[other]]fourth confidence level;

wherein generating the list of open ports comprises retrieving a previously constructed list of open ports.

25. (Original) The method of Claim 24, wherein the network node is one of a computer, a router, and a printer.

26. (Previously Presented) The method of Claim 24, wherein transmitting at least a first plurality of Internet Control Message Protocol packets further includes transmitting at least a first User Datagram Protocol packet to the network node and receiving in response at least a second User Datagram Protocol packet, and wherein the generated second identification and the second confidence level are based, in addition to the second plurality of Internet Control Message Protocol packets, on at least the second User Datagram Protocol packet.

27. (Cancelled)

28. (Previously Presented) The method of Claim 24, further comprising determining whether NULL session access is available on at least one port configured to run at least one of a Server Message Block service and a NETBIOS service, and if such NULL session access is available, using such NULL session access to determine at least a major version and a minor version of the operating system executed by the network node, and generating, based on the major version and the minor version, a fifth identification of which operating system is executed by the network node and a fifth confidence level indicating a degree to which the fifth identification is deemed accurate, wherein generating the overall identification of the operating system is further based on the fifth identification and the fifth confidence level.

29. (Previously Presented) The method of Claim 24, wherein generating overall identification of an operating system includes selecting as the overall identified operating system the operating system identified by one of the first identification, the second identification, the third identification, and the fourth identification.

30. (Cancelled)

31. (Previously Presented) The method of Claim 24, wherein the first plurality of Transmission Control Protocol packets are compliant with a specification of Transmission Control Protocol packets defined by DARPA Request for Comments 793.

32. (Previously Presented) A method of identifying an operating system executed by a network node, comprising:

executing a plurality of tests for identifying which operating system is executed by a network node, such that each test returns an identification of an operating system executed by the network node;

assessing, based at least on one characteristic of each identification of the operating system returned by the plurality of tests, which of the tests to select for determining an overall identification of the operating system; and

generating an overall identification of the operating system executed by the network node as the operating system that is identified by the selected test;

wherein a confidence level is assigned to the identification of the operating system based on a predetermined confidence level stored in association with at least one of a plurality of identification fingerprints used to identify the operating system;

wherein the identification of the operating system by one of the plurality of tests is dependent on the identification of the operating system by another one of the plurality of tests;

wherein a list of open ports on the network node is generated and, based on the list of open ports, another identification of which operating system is executed by the network node and another confidence level indicating a degree to which the other identification is deemed accurate are generated, wherein making the overall identification of the operating system is further based on the other identification and the other confidence level;

wherein generating the list of open ports comprises retrieving a previously constructed list of open ports.

33. (Original) The method of Claim 32, further comprising resolving conflicts among

identifications made by the plurality of tests, wherein the resolving conflicts is based at least in part on comparing aggregated results from at least two of the plurality of tests with a plurality of conflict resolution definitions.

34. (Original) The method of Claim 32, wherein each of the tests returns an identification of an operating system that is not influenced by the identification returned by any of the other tests.

35. (Original) The method of Claim 32, wherein the plurality of tests includes at least a first test in which the returned identification of an operating system is generated based on at least connecting to at least one open port on the network node and transmitting to the open port data configured to cause the open port to return at least one banner.

36. (Original) The method of Claim 35, wherein the plurality of tests further includes at least a second test in which the returned identification of an operating system is generated based on at least generating a list of open ports on the network node.

37. (Previously Presented) The method of Claim 36, wherein at least one characteristic of each operating system identification on which the assessing of a test to rely upon is based is the confidence level that each operating system identification is correct.

38. (Original) The method of Claim 37, wherein at least one confidence level concerning whether an operating system identification is correct is determined using a fitness calculation.

39. (Previously Presented) A method of identifying an operating system executed by a network node, comprising:

executing a plurality of tests for identifying which operating system is executed by a network node, each test producing actual test results indicative of at least an identification of an operating system executed by the network node;

determining that at least one of the plurality of tests have actual test results that disagree about which operating system is executed by the network node;

deriving, from the plurality of actual test results, a group of aggregate actual test results that includes at least a portion of at least two of the plurality of actual test results;

comparing the group of aggregate actual test results with a plurality of conflict resolution definitions and finding a closest match between the group of aggregate actual test results and the conflict resolution definitions, wherein each conflict resolution definition is associated with an operating system that is deemed to be the operating system being executed by the network node; and

making an overall identification of the operating system executed by the network node, wherein the overall identified operating system is deemed to be the operating system associated with the closest matched conflict resolution definition;

wherein a confidence level is assigned to the identification of the operating system, based on a predetermined confidence level stored in association with at least one of the plurality of tests used to identify the operating system;

wherein the identification of the operating system by of the plurality of tests is dependent on the identification of the operating system by another one of the plurality of tests;

wherein a list of open ports on the network node is generated and, based on the list of open ports, another identification of which operating system is executed by the network node and another confidence level indicating a degree to which the other identification is deemed accurate are generated, wherein making the overall identification of the operating system is further based on the other identification and the other confidence level;

wherein generating the list of open ports comprises retrieving a previously constructed list of open ports.

40. (Previously Presented) The method of Claim 39, wherein the actual test results are further indicative of the confidence level indicating a degree to which the identification of an operating system executed by the network node is accurate.

41. (Original) The method of Claim 39, wherein the plurality of tests includes a first test comprising transmitting a first plurality of Transmission Control Protocol packets to a network node on a computer network, receiving in response a second plurality of Transmission Control Protocol packets, and generating, based on characteristics of the second plurality of Transmission Control Protocol packets, a first identification of which operating system is executed by the network node.

42. (Original) The method of Claim 41, wherein the plurality of tests further includes a second test comprising transmitting at least a first plurality of Internet Control Message Protocol packets to the network node, receiving in response at least a second plurality of Internet Control Message Protocol packets, and generating, based at least on characteristics of the second plurality of Internet Control Message Protocol packets, a second determination of which operating system is executed by the network node.

43. (Original) The method of Claim 42, wherein the plurality of tests further includes a third test comprising connecting to at least one open port on the network node, transmitting to the open port data configured to cause the open port to return at least one banner, and generating, based on the at least one banner, a third determination of which operating system is executed by the network node.

44. (Previously Presented) The method of Claim 43, wherein the plurality of tests further includes a fourth test comprising generating the list of open ports on the network node and generating, based on the list of open ports, a fourth determination of which operating system is executed by the network node.

45. (Original) The method of Claim 44, wherein the plurality of tests further includes a fifth test comprising determining whether NULL session access is available on at least one port configured to run at least one of a Server Message Block service and a NETBIOS service, and if such NULL session access is available, using such NULL session access to determine at least a major version and a minor version of the operating system executed by the network node, and generating, based on the major version and the

minor version, a fifth determination of which operating system is executed by the network node.

46. (Previously Presented) The operating system identification system of Claim 1, wherein each operating system identification test executed by the identification module causes a first plurality of packets to be transmitted to the network node and a plurality of response packets to be received by each operating system identification test.

47. (Previously Presented) The operating system identification system of Claim 46, wherein the plurality of response packets are reformatted for use in identifying the operating system being executed by the network node.

48. (Previously Presented) The operating system identification system of Claim 1, further comprising resolving conflicts among the at least one of the identifications made by the plurality of operating system identification tests only if none of the at least one of the identifications is associated with the confidence level greater than the predetermined confidence level.

49. (Previously Presented) The operating system identification system of Claim 48, wherein the resolving conflicts is based at least in part on comparing aggregated results from at least two of the plurality of operating system identification tests with a plurality of conflict resolution definitions.